МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”

**КАФЕДРА ИИТ**

ОТЧЁТ

по лабораторной работе №3

**«Классы и объекты в С++»**

Выполнил:

Студент 1 курса

группы ПО-9

Харитонович Захар Сергеевич

Проверил:

Скарубо А. О.

Брест 2022

**Цель работы:** Получить практические навыки реализации классов на языке С++.

**Порядок выполнения работы**

*Вариант 10*

*Задание*

Определить пользовательский класс в соответствии с вариантом задания.

Класс «Животное»

имя – char\*

класс – char\*

средний вес – int

Определить в классе конструкторы без параметров, с параметрами, копирования. Определить в классе деструктор. Определить в классе компоненты-функции для просмотра и установки полей данных.

Написать демонстрационную программу, в которой создаются и разрушаются объекты пользовательского класса и каждый вызов конструк-тора и деструктора сопровождается выдачей соответствующего сообщения (какой объект какой конструктор или деструктор вызвал).

Показать в программе использование указателя на объект.

**Определение пользовательского класса** (заголовочный файл *animal.h*):

#ifndef ANIMAL\_H

#define ANIMAL\_H

class Animal {

public:

Animal(); // конструктор без параметров

Animal(const char \*, const char \*, const int); // конструктор с параметрами

Animal(const Animal &); // конструктор копирования

~Animal(); // деструктор

char \* GetName(); // геттер для поля name

char \* GetBioClass(); //геттер для поля bioClass

int GetAvWeight(); // геттер для поля avWeight

void SetName(const char \*); // сеттер для поля name

void SetBioClass(const char \*); // сеттер для поля bioClass

void SetAvWeight(const int); // сеттер для поля avWeight

void Display(); // вывод содержимого класса

private:

char name[256] = "default"; // имя

char bioClass[256] = "default"; // класс (биологический)

int avWeight = 0; // средняя масса

};

#endif

**Реализация класса** (файл *animal.cpp*):

#include "Animal.h"

#include <iostream>

#include <string.h>

Animal::Animal() {

std::cout << "Object " << this << " created." << std::endl;

}

Animal::Animal(const char \* curName, const char \* curClass, const int curWeight): avWeight(curWeight) {

strcpy(name, curName);

strcpy(bioClass, curClass);

std::cout << "Object " << this << " created with parameters." << std::endl;

}

Animal::Animal(const Animal & current) {

strcpy(name, current.name);

strcpy(bioClass, current.bioClass);

avWeight = current.avWeight;

std::cout << "Object " << this << " created with copying." << std::endl;

}

Animal::~Animal() {

std::cout << "Object " << this << " deleted." << std::endl;

}

char \* Animal::GetName() {

return name;

}

char \* Animal::GetBioClass() {

return bioClass;

}

int Animal::GetAvWeight() {

return avWeight;

}

void Animal::SetName(const char \* curName) {

strcpy(name, curName);

}

void Animal::SetBioClass(const char \* curClass) {

strcpy(bioClass, curClass);

}

void Animal::SetAvWeight(const int curWeight) {

avWeight = curWeight;

}

void Animal::Display() {

std::cout << "Name: " << name << "; Class: " << bioClass << "; Average weight: " << avWeight << " kg." << std::endl;

}

**Демонстрационная программа** (файл *main.cpp*):

#include <iostream>

#include "animal.h"

using namespace std;

int main() {

Animal bison;

bison.Display();

char name[] = "Bison", bioClass[] = "Mammals";

bison.SetName(name);

bison.SetBioClass(bioClass);

bison.SetAvWeight(700);

bison.Display();

Animal wolf("Wolf", "Mammals", 40);

wolf.Display();

Animal wolfClone(wolf);

wolfClone.Display();

Animal bisonClone = bison;

bisonClone.Display();

Animal \* bison\_ptr = &bison;

bison\_ptr->SetName("Bison bonasus");

bison\_ptr->Display();

return 0;

}

**Вывод демонстрационной программы:**

